APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
09/714,326	11/16/2000	Janin Pascal	00-GR-241	4342

23334   7590   04/07/2004

FLEIT, KAIN, GIBBONS, GUTMAN, BONGINI
& BIANCO P.L.
ONE BOCA COMMERCE CENTER
551 NORTHWEST 77TH STREET, SUITE 111
BOCA RATON, FL 33487

EXAMINER
IQBAL, NADEEM

ART UNIT	PAPER NUMBER
2114	

DATE MAILED: 04/07/2004

Please find below and/or attached an Office communication concerning this application or proceeding.

Office Action Summary	Application No 09/714,326	Applicant(s) PASCAL, JANIN
	Examiner Nadeem Iqbal	Art Unit 2114

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE <u>3</u> MONTH(S) FROM
THE MAILING DATE OF THIS COMMUNICATION.
- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed
  after SIX (6) MONTHS from the mailing date of this communication.
- If the period for reply specified above is less than thirty (30) days, a reply within the statutory minimum of thirty (30) days will be considered timely.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133).
  Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any
  earned patent term adjustment. See 37 CFR 1.704(b).

Status

1) ☒ Responsive to communication(s) filed on *28 January 2004*.

2a) ☐ This action is **FINAL**.   2b) ☒ This action is non-final.

3) ☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is
closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

4) ☒ Claim(s) *1-47* is/are pending in the application.

    4a) Of the above claim(s) _____ is/are withdrawn from consideration.

5) ☐ Claim(s) _____ is/are allowed.

6) ☐ Claim(s) *1-4,7-9,12-18,21-23,25-31,34-36 and 39-47* is/are rejected.

7) ☒ Claim(s) *5,6,10,11,19,20,24,32,33,37 and 38* is/are objected to.

8) ☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

9) ☐ The specification is objected to by the Examiner.

10) ☐ The drawing(s) filed on _____ is/are: a)☐ accepted or b)☐ objected to by the Examiner.

    Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).

    Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).

11) ☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

12) ☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).

    a)☐ All b)☐ Some * c)☐ None of:

      1.☐ Certified copies of the priority documents have been received.

      2.☐ Certified copies of the priority documents have been received in Application No. _____.

      3.☐ Copies of the certified copies of the priority documents have been received in this National Stage
application from the International Bureau (PCT Rule 17.2(a)).

    * See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

1) ☒ Notice of References Cited (PTO-892)

2) ☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)

3) ☐ Information Disclosure Statement(s) (PTO-1449 or PTO/SB/08)
Paper No(s)/Mail Date _____.

4) ☐ Interview Summary (PTO-413)
Paper No(s)/Mail Date. _____.

5) ☐ Notice of Informal Patent Application (PTO-152)

6) ☐ Other: _____.

Response to Amendment

This office action is responsive to an amendment filed on Jan. 28, 2004.

Claim Rejections - 35 USC § 103

1.   The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all

obviousness rejections set forth in this Office action:

> (a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in
> section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are
> such that the subject matter as a whole would have been obvious at the time the invention was made to a person
> having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negatived by the
> manner in which the invention was made.

2.   Claims 1-4, & 7 are rejected under 35 U.S.C. 103(a) as being unpatentable over Bates et

al., (U.S. Patent Number 6587967).

3.   As per claim 1, Bates et al., teaches (col. 2, lines 43-45) a program product and method of

executing a portion of a multithreaded program and monitoring threads that are executing within

a selected monitored region in the program while the portion of the program is executing. He

thus teaches limitations pertain to monitoring the activation of programmed sequences of

programmed system that comprises at least a first and a second programmed sequence, each to

be executed iteratively. He also teaches (col. 2, lines 60-63) that threads executing within a

monitored region may be highlighted while program execution continues. He also teaches (col. 3,

lines 44-46) that a user can specify one or more sections of one of more computer programs for

which thread activity is to be monitored. He thus teaches first programmed sequence is made to

monitor the execution of the second programmed sequence. He does not explicitly disclose the

second programmed sequence is made to monitor the first programmed sequence. It would have

been obvious to a person of ordinary skill in the art at the time the invention was made to realize

that He clearly provides the capability to monitor the first programmed sequence to monitor the

execution of the second programmed sequence, and second programmed sequence is made to

monitor the first programmed sequence. This is because he teaches as stated above that a user

can specify one or more sections of one of more computer programs for which thread activity is

to be monitored, and also teaches (col. 2, lines 66, 67) that the threads to be monitored may be

filtered, such as by specifying threads to be included or excluded, therefore clearly allows one to

specify the first programmed sequence is made to monitor the execution of the second

programmed sequence and the second programmed sequence is made to monitor the first

programmed sequence.

4.   As per claim 2, He teaches as stated per claim 1 above a program product and method of

executing a portion of a multithreaded program and monitoring threads that are executing within

a selected monitored region in the program while the portion of the program is executing, thus

would include a group consisting of routines and main program loops.

5.   As per claims 3 & 4, He teaches (col. 5, lines 7-10) illustrating a thread monitoring

capability in block diagram form, with the elements shown that contribute to maintaining control

points and to responding to a system exception. He also shows (col. 5, lines 14-16) a debugger

command is made setting a monitored region or a breakpoint. He thus clearly teaches

programmed sequences to include at least one interrupt routine.

Allowable Subject Matter

6.   Claims 5, 6, 10, 11, 19, 20, 24, 32, 33, 37 & 38 are objected to as being dependent upon a

rejected base claim, but would be allowable if rewritten in independent form including all of the

limitations of the base claim and any intervening claims.

7.   As per claim 7, He teaches (col. 5, lines 56-58) a breakpoint manager routine for

maintaining control points and responding to a system exception, thus would allow for a failure

due to activation of a programmed sequence to cause a complete or partial reset of the

programmed system.

8.   Claims 8, 9, 12-18, 21-23, 25-31, 34-36, 39-47 are rejected under 35 U.S.C. 103(a) as

being unpatentable over Bates et al., (U.S. Patent Number 6587967).

9.   As per claims 8 & 22, 27, Bates et al., substantially teaches the claimed invention as

disclosed related to claim 1 above. He also teaches (col. 2, lines 43-45) a method of executing a

portion of a multithreaded program and monitoring threads that are executing within a selected

monitored region in the program while the portion of the program is executing. He thus teaches

limitations pertain to monitoring the activation of programmed sequences of programmed system

that comprises at least a first and a second programmed sequence, each to be executed

iteratively. He also teaches (col. 2, lines 60-63) that threads executing within a monitored region

may be highlighted while program execution continues. He also teaches (col. 3, lines 44-46) that

a user can specify one or more sections of one of more computer programs for which thread

activity is to be monitored. He thus teaches first programmed sequence is made to monitor the

execution of the second programmed sequence. He does not explicitly disclose N programmed

sequences is monitored by at least one other programmed sequence. It would have been obvious

to a person of ordinary skill in the art to realize that He clearly provides the capability to monitor

the N programmed sequences by at least one other programmed sequence. This is because he

teaches as stated above that a user can specify one or more sections of one of more computer

programs for which thread activity is to be monitored, and also teaches (col. 2, lines 66, 67) that

the threads to be monitored may be filtered, such as by specifying threads to be included or

excluded, therefore clearly would allows N programmed sequences to be monitored by at least

one other programmed sequence.

10.   As per claims 9, 23 & 46, He teaches (col. 2, lines 60-63) that threads executing within a

monitored region may be highlighted while program execution continues. He also teaches (col. 3,

lines 44-46) that a user can specify one or more sections of one of more computer programs for

which thread activity is to be monitored, therefore would allow the N programmed sequence to

be monitored by each of the N-1 other programmed sequences.

11.   As per claims 12 & 13, He teaches (col. 5, lines 7-10) illustrating a thread monitoring

capability in block diagram form, with the elements shown that contribute to maintaining control

points and to responding to a system exception. He also shows (col. 5, lines 14-16) a debugger

command is made setting a monitored region or a breakpoint. He thus clearly teaches

programmed sequences to include at least one interrupt routine.

12.   As per claims 14 & 15, He teaches (col. 5, lines 56-58) a breakpoint manager routine for

maintaining control points and responding to a system exception, thus would allow for a failure

due to activation of a programmed sequence to cause a complete or partial reset of the

programmed system.

13.   As per claims 16, 21, 28, Bates substantially teaches the claimed invention as disclosed

related to claim 1 above. He also teaches (col. 2, lines 43-45) a program product and method of

executing a portion of a multithreaded program and monitoring threads that are executing within

a selected monitored region in the program while the portion of the program is executing. He

thus teaches limitations pertain to monitoring the activation of programmed sequences of

programmed system that comprises at least a first and a second programmed sequence, each to

be executed iteratively. He also teaches (col. 2, lines 60-63) that threads executing within a

monitored region may be highlighted while program execution continues. He also teaches (col. 3,

lines 44-46) that a user can specify one or more sections of one of more computer programs for

which thread activity is to be monitored. He thus teaches first programmed sequence is made to

monitor the execution of the second programmed sequence. He does not explicitly disclose the

second programmed sequence is made to monitor the first programmed sequence. It would have

been obvious to a person of ordinary skill in the art to realize that He clearly provides the

capability to monitor the first programmed sequence to monitor the execution of the second

programmed sequence, and second programmed sequence is made to monitor the first

programmed sequence. This is because he teaches as stated above that a user can specify one or

more sections of one of more computer programs for which thread activity is to be monitored,

and also teaches (col. 2, lines 66, 67) that the threads to be monitored may be filtered, such as by

specifying threads to be included or excluded, therefore clearly allows one to specify the first

programmed sequence is made to monitor the execution of the second programmed sequence and

the second programmed sequence is made to monitor the first programmed sequence.

14.   As per claims 17, 29 He teaches as stated per claim 1 above a program product and

method of executing a portion of a multithreaded program and monitoring threads that are

executing within a selected monitored region in the program while the portion of the program is

executing, thus would include a group consisting of routines and main program loops.

15.   As per claims 18, 26, 30 & 31, He teaches (col. 5, lines 7-10) illustrating a thread

monitoring capability in block diagram form, with the elements shown that contribute to

maintaining control points and to responding to a system exception. He also shows (col. 5, lines

14-16) a debugger command is made setting a monitored region or a breakpoint. He thus clearly

teaches programmed sequences to include at least one interrupt routine.

16.   As per claim 25, He teaches as stated per claim 1 above a program product and method of

executing a portion of a multithreaded program and monitoring threads that are executing within

a selected monitored region in the program while the portion of the program is executing, thus

would include a group consisting of routines and main program loops.

17.   As per claim 34, He teaches (col. 5, lines 56-58) a breakpoint manager routine for

maintaining control points and responding to a system exception, thus would allow for a failure

due to activation of a programmed sequence to cause a complete or partial reset of the

programmed system.

18.   As per claims 35 & 43, Bates et al., substantially teaches the claimed invention as

disclosed related to claim 8 above. He also teaches (col. 2, lines 43-45) a method of executing a

portion of a multithreaded program and monitoring threads that are executing within a selected

monitored region in the program while the portion of the program is executing. He thus teaches

limitations pertain to monitoring the activation of programmed sequences of programmed system

that comprises at least a first and a second programmed sequence, each to be executed

iteratively. He also teaches (col. 2, lines 60-63) that threads executing within a monitored region

may be highlighted while program execution continues. He also teaches (col. 3, lines 44-46) that

a user can specify one or more sections of one of more computer programs for which thread

activity is to be monitored. He thus teaches first programmed sequence is made to monitor the

execution of the second programmed sequence. He does not explicitly disclose N programmed

sequences is monitored by at least one of the N-1 other programmed sequence. It would have

been obvious to a person of ordinary skill in the art to realize that He clearly provides the

capability to monitor the N programmed sequences by at least one other programmed sequence.

This is because he teaches as stated above that a user can specify one or more sections of one of

more computer programs for which thread activity is to be monitored, and also teaches (col. 2,

lines 66, 67) that the threads to be monitored may be filtered, such as by specifying threads to be

included or excluded, therefore clearly would allows N programmed sequences to be monitored

by at least one other programmed sequence.

19.   As per claims 36, 44, 45 & 47, He teaches (col. 2, lines 60-63) that threads executing

within a monitored region may be highlighted while program execution continues. He also

teaches (col. 3, lines 44-46) that a user can specify one or more sections of one of more computer

programs for which thread activity is to be monitored, therefore would allow the N programmed

sequence to be monitored by each of the N-1 other programmed sequences.

20.   As per claim 39, He teaches as stated per claim 1 above a program product and method of

executing a portion of a multithreaded program and monitoring threads that are executing within

a selected monitored region in the program while the portion of the program is executing, thus

would include a group consisting of routines and main program loops.

21.   As per claims 40 & 41, He teaches (col. 5, lines 7-10) illustrating a thread monitoring

capability in block diagram form, with the elements shown that contribute to maintaining control

points and to responding to a system exception. He also shows (col. 5, lines 14-16) a debugger

command is made setting a monitored region or a breakpoint. He thus clearly teaches

programmed sequences to include at least one interrupt routine.

22.   As per claim 42, He teaches (col. 5, lines 56-58) a breakpoint manager routine for

maintaining control points and responding to a system exception, thus would allow for a failure

due to activation of a programmed sequence to cause a complete or partial reset of the
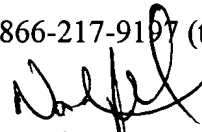
programmed system.

Response to Arguments

23.   Applicant's arguments with respect to claims 1-47 have been considered but are moot in

view of the new ground(s) of rejection.

Conclusion

Any inquiry concerning this communication or earlier communications from the

examiner should be directed to Nadeem Iqbal whose telephone number is (703)-308-5228. The

examiner can normally be reached on M-F (8:00-5:30) First Friday Off.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's

supervisor, Robert W Beausoliel can be reached on (703)-305-9713. The fax phone number for

the organization where this application or proceeding is assigned is 703-872-9306.

Information regarding the status of an application may be obtained from the Patent

Application Information Retrieval (PAIR) system. Status information for published applications

may be obtained from either Private PAIR or Public PAIR. Status information for unpublished

applications is available through Private PAIR only. For more information about the PAIR

system, see http://pair-direct.uspto.gov. Should you have questions on access to the Private PAIR

system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free).

Nadeem Iqbal
Primary Examiner
Art Unit 2114

NI